# Expected heights in heaps

Jeannette M. de Graaf and Walter A. Kosters

*Department of Mathematics and Computer Science*
*Leiden University*
*P.O. Box 9512*
*2300 RA Leiden*
*The Netherlands*

**Abstract** In this paper several recurrences and formulas are presented leading to an upper bound and a lower bound, both logarithmic, for the expected height of a node in a heap. These bounds are of interest for algorithms that select the $k$th smallest element in a heap.

**CR categories**: E.1, F.2.2, G.2.1.

**Keywords**: expected height, heap, recurrences, upper and lower bounds.

This paper was published in BIT **32** (1992), 570–579.

## 1. Introduction

In this paper we consider heaps. A heap is a binary tree, in which every level is completely filled with nodes, except for (perhaps) the lowest one. At this lowest level all nodes are as far to the left as possible. We assume that all nodes contain a unique integer. A heap has to satisfy the 'heap-property': every node, except for the root, contains an integer that is larger than the integer in its father. Such a heap is sometimes called a min-heap. In the literature 'larger' is often replaced by 'smaller', but this gives rise to an entirely equivalent notion of heap (max-heap).

In [3] a logarithmic upper bound for the expected height of a node in a heap is derived. This bound is used for the analysis of an algorithm that selects the $k$th smallest element in a heap. Another algorithm for this problem is presented in [1], together with some comments on previous results. In our paper we shall present several new recurrences and formulas related to these problems. In particular the upper bound follows as an easy consequence. In [3] there are some conjectures concerning a lower bound. In Section 4 we shall give a logarithmic lower bound, thereby settling one of these conjectures. A more precise description of the organization of this paper is given at the end of this section.

The height of a node in a tree $T$ (denoted by $ht(T, node)$) is defined as the number of nodes on the (shortest) path from the root to that particular node. So the root has height 1. Sometimes one defines the height of a node as the number of edges on this path, which is one less than the number of nodes.

If a heap contains $n$ nodes ($n > 0$) we may—for simplicity—assume that the integers in the tree are exactly those in $\{1, 2, \ldots, n\}$. Of course, 1 is in the root. From now on we identify nodes and the numbers they contain, so 1 *is* the root of the heap.

Suppose that an integer $k$ is given with $1 \leq k \leq n$. We want to know the height of $k$. Of course, when $k = 1$, this height equals 1; for $k = 2$ we get 2. In order to compute the probability that $k$ is at height $i$ we have to define the probability that a certain heap arises. It seems natural to impose:

**Assumption A.** *All heaps on the numbers* $\{1, 2, \ldots, n\}$ *are equally likely.*

If one assumes that all permutations of $\{1, 2, \ldots, n\}$ are equally likely, and one uses the usual Williams-method of heapconstruction, Assumption A is satisfied ([2], p. 155).

We are interested in the probability that node $k$ is at height $i$ in a heap on $\{1, 2, \ldots, n\}$, where $1 \leq i \leq k \leq n$. This probability is denoted by $P(k, i, n)$. We shall always assume that $n$ is sufficiently large, so that nodes of height $i$ do occur in the heap ($i \leq \lfloor {}^2log(n) \rfloor + 1$). Of course we have $P(1, i, n) = \delta_{i1}$, $P(2, i, n) = \delta_{i2}$ and $P(k, 1, n) = \delta_{k1}$. Here we used the Kronecker-delta $\delta_{ij}$, which equals 1 if $i = j$ and 0 otherwise.

It seems essential for our computations that the lowest level of the heap is completely filled with nodes. So from now on we suppose:

**Assumption B.** $n = 2^t - 1$ *for some integer* $t$ with $t \geq 1$.

Let $\lim'_{n\to\infty} P(k,i,n)$ denote the limit of $P(k,i,n)$ as $n$ tends to $\infty$ (where $'$ reminds of the fact that $n$ has to satisfy Assumption B), if this limit exists. The limit is denoted by $P(k,i)$. Of course we have $P(1,i) = \delta_{i1}$, $P(2,i) = \delta_{i2}$ and $P(k,1) = \delta_{k1}$.

The paper is organized in the following way. In Section 2 we derive a recurrence for $P(k,i,n)$ and show that $\lim'_{n\to\infty} P(k,i,n)$ exists. This leads to a recurrence for the expected height of the $k$th smallest element in a heap (where $n \to \infty$). Section 3 contains another approach also leading to these recurrences. Several other interesting formulas for $P(k,i,n)$ and $P(k,i)$ are given there. The recurrence for the expected height of $k$ is used in Section 4 to prove an upper bound $1 + {}^2log(k)$ and a lower bound ${}^2log(k)$ for this height. The lower bound (Theorem 8) requires—as far as we can see—a rather intricate proof.

## 2. Some recurrences

In this section we show that the $P(k,i,n)$, the $P(k,i)$ and the $E(k)$ (where $E(k)$ is the expected height of node $k$ in the case where we let $n$ tend to $\infty$) satisfy certain recurrences. The recurrence for $E(k)$ will later be used to prove upper and lower bounds.

**Theorem 1.** *For $2 \leq i \leq k \leq n$ and $i \leq {}^2log(n+1)$ we have:*

$$(*) \qquad P(k,i,n) = \sum_{j=i-1}^{k-1} \binom{k-2}{j-1} A(k,j,n) P(j, i-1, \frac{n-1}{2})$$

*where $A(k,j,n)$ is defined by:*

$$A(k,j,n) = \frac{2(\frac{n-1}{2})(\frac{n-1}{2}-1)\ldots(\frac{n-1}{2}-j+1)(\frac{n-1}{2})(\frac{n-1}{2}-1)\ldots(\frac{n-1}{2}-k+j+2)}{(n-1)(n-2)\ldots(n-k+1)}$$

*Proof.* Let $Q(k,j,n)$ be the probability that $k$ is the $j$th smallest element in the right subtree of the root, given that $k$ indeed occurs in this subtree. Note that this subtree has $\frac{n-1}{2}$ nodes. We can compute $Q(k,j,n)$ as follows. The smallest key is always in the root. The $k$th smallest key in the heap is the $j$th smallest key in the right subtree for some number $j$. So from the remaining $k-2$ keys larger than 1 and smaller than $k$, we must choose $j-1$ in the right subtree. This can be done in $\binom{k-2}{j-1}$ ways. From the remaining $n-k$ keys we must then choose $(n-1)/2-j$ in the right subtree. This kan be done in $\binom{n-k}{\frac{n-1}{2}-j}$ ways. Hence

$$Q(k,j,n) = \frac{\binom{k-2}{j-1}\binom{n-k}{\frac{n-1}{2}-j}}{\binom{n-2}{\frac{n-1}{2}}}$$

where in the denominator a factor $n-2$ arises since we have assumed $k$ in the right subtree. A similar result holds of course for the left subtree. The probability that the $k$th key is in the left subtree is equal to the probability that it is in the right subtree. Therefore the probability that in a heap the $k$th smallest key is the $j$th smallest in one of the two subheaps is given by:

$$\frac{\binom{k-2}{j-1}\binom{n-k}{\frac{n-1}{2}-j}}{\binom{n-2}{\frac{n-1}{2}}} = \binom{k-2}{j-1} A(k,j,n)$$

Now the formula for $P(k,i,n)$ follows. |

**Corollary 2.** *For $2 \leq i \leq k$ we have that $P(k,i) = \lim'_{n\to\infty} P(k,i,n)$ exists and*

$$P(k,i) = (1/2)^{k-2} \sum_{j=i-1}^{k-1} \binom{k-2}{j-1} P(j, i-1)$$

*Proof.* From the definition of $A(k,j,n)$ we deduce: $\lim'_{n\to\infty} A(k,j,n)$ exists and equals $(1/2)^{k-2}$. Using $(*)$ one can now easily prove (by induction with respect to $i$) that $\lim'_{n\to\infty} P(k,i,n)$ exists. Finally, if one takes the limit in $(*)$, the recurrence for $P(k,i)$ follows immediately. In the next section we shall give another proof of this corollary. |

Using this recurrence it is rather easy to compute $P(k,i)$ for small values of $i$. For instance:

$$P(k,2) = (1/2)^{k-2} \qquad (k \geq 2)$$
$$P(k,3) = 2(3/4)^{k-2} - 2(1/2)^{k-2} \qquad (k \geq 3)$$
$$P(k,4) = \frac{8}{3}(7/8)^{k-2} - 4(3/4)^{k-2} + \frac{4}{3}(1/2)^{k-2} \qquad (k \geq 4)$$

We are particularly interested in the expected height $E_n(k)$ of $k$. $E_n(k)$ is given by $\sum_{i=1}^{k} iP(k,i,n)$ for $k \leq n$. Now we let $E(k) = \lim'_{n\to\infty} E_n(k)$. This limit exists (since $\lim'_{n\to\infty} P(k,i,n)$ exists) and equals $\sum_{i=1}^{k} iP(k,i)$. Obviously $E(1) = 1$, and using the result derived in Corollary 2 we find, after interchanging the order of the summations involved:

**Corollary 3.** *We have:*

$$\begin{cases} E(1) = 1 \\ E(k) = 1 + (1/2)^{k-2} \sum_{i=0}^{k-2} \binom{k-2}{i} E(i+1) \qquad (k \geq 2) \end{cases}$$ |

**Remark.** If Assumption B is not satisfied the limit of $P(k,i,n)$ does not exist in general. For example one can show that for arbitrary $n > 2$ (without Assumption B):

$$P(3,2,n) = \frac{2}{(n-1)(n-2)}\left(2^{\lfloor {}^2log(n)\rfloor - \delta(n)} - 1\right)\left(n - 2^{\lfloor {}^2log(n)\rfloor - \delta(n)}\right)$$

where

$$\delta(n) = \begin{cases} 1 & \text{if } 2^t \leq n \leq 2^t + 2^{t-1} - 1 \text{ for some integer } t > 0 \\ 0 & \text{otherwise} \end{cases}$$

Imposing suitable restrictions on $n$ one can achieve any value between 5/2 and 23/9 as a limit for $E_n(3)$. (In the case where $n$ satisfies Assumption B, Corollary 3 yields $E(3) = 5/2$.) However, if one weakens the definition of 'heap' in such a way that the lowest level is not necessarily filled to the left, it might be possible that the limit exists (averaging over all possible shapes of such heaps). Another possibility would be to consider ordinary binary trees with the 'heap-property'. We have not studied these situations in detail yet.

## 3. Some other formulas

In this section is we shall follow another line of reasoning, also leading to the result mentioned in Corollary 2. The proofs in this section are a bit more combinatorial.

**Theorem 4.** *For $1 \le i \le k \le n$ and $i \le {}^2log(n+1)$ we have:*

$$P(k,i,n) = \frac{1}{n(n-1)\dots(n-k+1)} \sum_T \prod_{j=1}^{k} \left( \frac{n - (2^{ht(T,j)-1} - 1)}{2^{ht(T,j)-1}} \right)$$

*In this summation $T$ runs over all binary* trees *that contain only $\{1,2,\dots,k\}$, satisfy the 'heap-property', and have $k$ at height $i$. If $n$ is small we restrict the summation to those $T$ that arise as subtree of the original heap. (Otherwise, some trees would be higher than this heap.)*

*Proof.* From [2], p.154, we know that the number of heaps containing $\{1,2,\dots,n\}$ equals $n!$ divided by the product over all subheaps of the number of nodes in these subheaps. The number of nodes in a subheap rooted at a node at height $j$ equals (with $t = {}^2log(n+1)$)

$$2^{t-j+1} - 1 = \frac{n - (2^{j-1} - 1)}{2^{j-1}}$$

We compute $P(k,i,n)$ as the sum over $T$ of (the number of heaps on $\{1,2,\dots,n\}$ where the numbers $\{1,2,\dots,k\}$ are situated as they are in $T$) divided by (the total number of heaps on $\{1,2,\dots,n\}$).
An easy argument shows that this summation boils down to the formula in the theorem. Indeed, once $T$ is given, we only have to spread the numbers $\{k+1,\dots,n\}$ over the remaining $n-k$ nodes in the heap (maintaining the 'heap-property'). This gives rise to a factor $(n-k)!$ divided by the product (over the subheaps rooted at $\{k+1,\dots,n\}$) of the number of nodes in the subheaps. These terms cancel out, and finally only the subheaps rooted at $\{1,2,\dots,k\}$ contribute to the sum. |

Since—for $k$ and $i$ fixed—numerator and denominator in Theorem 4 are both polynomials in $n$ of degree $k$, we can conclude (cf. Corollary 2):

**Theorem 5.** *For $1 \le i \le k$ we have that $P(k,i) = \lim'_{n\to\infty} P(k,i,n)$ exists and*

$$P(k,i) = \sum_T 1/2^{Pl(T)}$$

*where $T$ runs over all binary* trees *that contain only $\{1,2,\dots,k\}$, satisfy the 'heap-property', and have $k$ at height $i$. Here we defined $Pl(T)$ as the sum (over all nodes in $T$) of the lengths of the paths from the root to these nodes.* |

From Theorem 5 we can derive

**Theorem 6.** *For $1 \leq i \leq k$ we have:*

$$P(k,i) = \frac{(k-2)!}{2^{(i-1)(i-2)/2}} \sum_{\substack{t_1,\ldots,t_{i-1} \\ t_1+\cdots+t_{i-1}=k-i}} \frac{2^{-t_1-\cdots-(i-1)t_{i-1}}}{t_1!\ldots t_{i-1}!(t_{i-1}+1)\ldots(t_{i-1}+\cdots+t_2+i-2)}$$

*Proof.* We have to sum over all trees $T$ with $k$ at height $i$. The path from the root to $k$ contains exactly $i$ nodes. The numbers on this path are denoted by $a_1$, $a_2$, ..., $a_i$, with $a_1 = 1$ and $a_i = k$. Let $t_j$ be the number of nodes in the subtree rooted at $a_j$ $(j = 1, 2, \ldots, i-1)$. (This subtree is the subtree that does not contain $k$.) Together, these subtrees contain $k - i$ numbers.
Now we enumerate all possible trees $T$. First, choose a partition $t_1 + \ldots + t_{i-1} = k - i$. Then choose $t_1$ numbers from $\{2, \ldots, k-1\}$; these numbers will be stored in the subtree $T_1$ of $a_1$. The number $a_2$ is now completely determined: it is the smallest remaining number. Continue in this way: choose $t_2$ numbers from the remaining $k - 3 - t_1$ to fill the subtree $T_2$ of $a_2$, and so on. This leads to a factor

$$\binom{k-2}{t_1}\binom{k-3-t_1}{t_2}\cdots\binom{t_{i-1}}{t_{i-1}}$$

This construction fixes the contents of the subtrees $T_j$. We still have to sum over all possible $T$'s corresponding to these choices. We must compute terms like $2^{-jt_j}/2^{Pl(T_j)}$. Only the numbers in $T_j$ are chosen, but still all binary trees satisfying the 'heap-property' are possible. (The term $2^{-jt_j}$ arises because a node in $T_j$ has height $j + ht(T_j, node)$ in $T$.) Note that

$$\sum_S 1/2^{Pl(S)} = 1$$

if one sums over all binary trees $S$ satisfying the 'heap-property' (this easily follows from Theorem 5 by summation over $i$). So the summation gives rise to a factor $2^{-jt_j}$.
Finally taking into account the numbers on the path from the root to $k$ (contributing a factor $2^{-i(i-1)/2}$), and multiplying by $2^{i-1}$ (the number of possible shapes of this path), one arrives at the formula in the theorem. |

**Remark.** The recurrence in Corollary 2 is an easy consequence of Theorem 6.

## 4. Upper and lower bounds

The following theorem is due to [3]; here we present a proof using the recurrence for $E(k)$ derived in Section 2.

**Theorem 7.** *For $k \geq 1$ we have:*

$$E(k) \leq 1 + {}^{2}log(k)$$

*Proof.* Use induction with respect to $k$, the case $k = 1$ being trivial. Then (note that $\sum_{j=0}^{k} \binom{k}{j} = 2^k$):

$$E(k) \leq 2 + (1/2)^{k-2} \sum_{i=0}^{k-2} \binom{k-2}{i} \, {}^{2}log(i+1)$$

Combining the terms with summation-index $(k-2)/2 - j$ and $(k-2)/2 + j$, and using

$$(k/2 - j)(k/2 + j) = (k/2)^2 - j^2 \leq (k/2)^2$$

one easily deduces the desired result. |

We can also derive a lower bound for $E(k)$.

**Theorem 8.** *For all $k \geq 1$:*

$$E(k) \geq \, {}^{2}log(k)$$

*Proof.* We define

$$H(j,k) = 1 + (1/2)^{k-j} \sum_{i=0}^{k-j} \binom{k-j}{i} E(i+j) \qquad (1 \leq j \leq k)$$

For $k \geq 2$ we have $E(k) = H(1, k-1)$. The $H(j,k)$ are completely determined by

$$H(1,1) = 2 \tag{a}$$
$$H(j,j) = 1 + H(1, j-1) \qquad (j \geq 2) \tag{b}$$
$$H(j,k) = (H(j+1,k) + H(j,k-1))/2 \qquad (1 \leq j < k) \tag{c}$$

These equations form an easy scheme for the computation of the $E(k)$. First we shall prove that

$$H(j,k) \geq \, {}^{2}log(j+k) \qquad (1 \leq j \leq k)$$

The proof is divided in five steps.

(1) For integers $K \geq 2$ we let

$$b_K = 1 - \frac{K}{2} + \frac{1}{2}\sqrt{K^2+4}$$
$$a_K = 1/\ {}^2log(b_K+1)$$

Note that $b_K \downarrow 1$ and $a_K \uparrow 1$ for $K \to \infty$. We have

$$(b_K(j+1)+k)(b_Kj+k-1) \geq (b_Kj+k)^2 \qquad (1 \leq j < k,\ k \geq K)$$

(since $b_K > 1$ this is equivalent to $b_K^2 + (K-2)b_K - K \geq 0$) and

$$1 + a_K\ {}^2log(b_K+j-1) \geq a_K\ {}^2log(b_Kj+j) \qquad (j \geq 2)$$

In order to guarantee these inequalities $a_K$ and $b_K$ are the best choices possible.

(2) Let $K = 2$. Then $b_K = \sqrt{2}$. We shall show that

$$H(j,k) \geq a_2\ {}^2log(b_2j+k) + c_2 \qquad (1 \leq j \leq k)$$

where $c_2 = 1$. We prove this inequality by induction with respect to $j$ and $k$, using the defining relations $(a)$, $(b)$ and $(c)$. To this end we have to show:

$$2 \geq a_2\ {}^2log(b_2+1) + c_2 \tag{$a'$}$$
$$1 + a_2\ {}^2log(b_2+j-1) \geq a_2\ {}^2log(b_2j+j) \qquad (j \geq 2) \tag{$b'$}$$
$$(b_2(j+1)+k)(b_2j+k-1) \geq (b_2j+k)^2 \qquad (1 \leq j < k) \tag{$c'$}$$

Using step (1) it is easy to check that these inequalities hold.

(3) Suppose that for some integer $K' \geq 2$ we have found $c_{K'}$ such that

$$(**) \qquad H(j,k) \geq a_{K'}\ {}^2log(b_{K'}j+k) + c_{K'} \qquad (1 \leq j \leq k)$$

We want to use this lower bound to derive a better one. In fact, we show that for any integer $K > K'$ we have

$$H(j,k) \geq a_K\ {}^2log(b_Kj+k) + c_K \qquad (1 \leq j \leq k)$$

where $c_K$ is defined by

$$c_K = c_{K'} + \min_{1 \leq j \leq k \leq K-1} \{a_{K'}\ {}^2log(b_{K'}j+k) - a_K\ {}^2log(b_Kj+k)\}$$

To prove this we use induction (cf. step (2)). The basis of the induction is here the following finite set of inequalities

$$H(j,k) \geq a_K\ {}^2log(b_Kj+k) + c_K \qquad (1 \leq j \leq k \leq K-1)$$

These inequalities are satisfied by $(**)$ and the choice of $c_K$, which again is 'optimal'. The other inequalities involved—corresponding to $(b')$ and $(c')$ in step (2), but now with $k \geq K$—are met by the definitions of $a_K$ and $b_K$, cf. step (1).

(4) Here we prove that

$$\min_{1 \leq j \leq k \leq K-1} \{a_{K'}\ {}^2log(b_{K'}j + k) - a_K\ {}^2log(b_K j + k)\}$$

is attained for $j = k = K - 1$. In fact, consider

$$h(x, y) = a_{K'}\ {}^2log(b_{K'}x + y) - a_K\ {}^2log(b_K x + y)$$

for real $x$ and $y$ with $0 \leq x \leq y \leq K - 1$ and $y \geq 1$. Derivation with respect to y, using $1 < b_K < b_{K'}$, shows that for fixed $x$ the function $h(x, y)$ is strictly decreasing in $y$. Therefore the minimum is attained for $y = K - 1$. So we let $g(x) = h(x, K - 1)$. Then we have $g(0) = g(K - 1) = (a_{K'} - a_K)\ {}^2log(K - 1)$. Derivation with respect to $x$ reveals that $g(x)$ has precisely one stationary point, which has to be in the interval $[0, K - 1]$.
In order to show that $g(x)$ attains its minimal value (in the interval $[0, K - 1]$) in $K - 1$ it suffices to show that $g'(0) > 0$. We compute:

$$g'(0) = \frac{1}{(K-1)}\{\frac{b_{K'}}{{}^elog(b_{K'} + 1)} - \frac{b_K}{{}^elog(b_K + 1)}\}$$

where $1 < b_K < b_{K'} \leq \sqrt{2}$. To show that $g'(0) > 0$, it is sufficient to prove that $h(x) = x/\ {}^elog(x + 1)$ is increasing on the interval $[1, \sqrt{2}]$. Taking the derivative this is equivalent to

$$l(x) = {}^elog(x + 1) - \frac{x}{x + 1} > 0 \qquad (1 \leq x \leq \sqrt{2})$$

which follows from $l(1) > 0$ and $l'(x) > 0$. This completes the argument.

(5) Now we take $K = 2^m$ for some integer $m \geq 2$ and we let $K' = K/2$. Then we may conclude from steps (2), (3) and (4) that

$$H(j, k) \geq a_K\ {}^2log(b_K j + k) + c_K \qquad (1 \leq j \leq k)$$

where

$$c_K = c_{K/2} + (a_{K/2} - a_K)\ {}^2log(K - 1) \geq c_{K/2} + (a_{K/2} - a_K)\ {}^2log(K)$$

Repeating this process we get

$$\begin{aligned} c_K &\geq 1 + (a_{K/2} - a_K)\ {}^2log(K) + (a_{K/4} - a_{K/2})\ {}^2log(K/2) + \ldots + (a_2 - a_4)\ {}^2log(4) \\ &= 1 - a_K\ {}^2log(K) + (a_{K/2} + \ldots + a_4 + a_2) + a_2 \end{aligned}$$

We can estimate $b_r$ and $a_r$ (for $2 \leq r \leq K$) as follows:

$$1 < b_r = 1 - \frac{r}{2} + \frac{1}{2}\sqrt{r^2+4} = 1 + \frac{1}{\frac{r}{2} + \frac{1}{2}\sqrt{r^2+4}} < 1 + 1/r$$

and

$$1 > a_r = 1/\ {}^2log(b_r + 1) > 1/\ {}^2log(2 + 1/r) > 1 - \frac{1}{2r\ {}^elog(2)}$$

The last inequality is derived by estimating the function $v(x) = 1/\ {}^2log(2 + x)$. To do so take the first and second term of the Taylor-expansion of $v$ and show that $v''(x) > 0$. Now we derive

$$c_K \geq 1 - \ {}^2log(K) + (\ {}^2log(K) - 1) - \frac{1}{2\ {}^elog(2)}(1/(K/2) + \ldots + 1/4 + 1/2) + a_2$$
$$\geq \frac{1}{{}^2log(\sqrt{2}+1)} - \frac{1}{2\ {}^elog(2)} \geq 0$$

For any integer $m \geq 1$ we have, with $K = 2^m$:

$$H(j,k) \geq \frac{{}^2log(b_K j + k)}{{}^2log(b_K + 1)} \qquad (1 \leq j \leq k)$$

If we let $K \to \infty$ we can conclude

$$H(j,k) \geq \ {}^2log(j + k) \qquad (1 \leq j \leq k)$$

hereby proving the promised result.

Finally we have

$$E(k) = H(1, k-1) \geq \ {}^2log(k) \qquad (k \geq 2)$$

This completes the proof of the theorem (the case $k = 1$ being trivial). |

**Remark.** Numerical results (up to $k = 5000$) suggest that

$$E(k) \geq \ {}^2log(k) + 0.72... \qquad (k \geq 1)$$

and

$$\lim_{k\to\infty} (E(k) - \ {}^2log(k)) = 0.72...$$

However, we were not able to prove this in general. The argument in the proof of Theorem 8 can be slightly improved, but we could not do better than a constant 0.34 instead of the constant 0.72... mentioned above. The lower bound for $E(k)$ (apart from the constant involved) is also conjectured in [3].

## Acknowledgements

The authors are very grateful to Henk J.M. Goeman for his stimulating encouragements and discussions, and to Hendrik Jan Hoogeboom for his careful reading of the manuscript.

## References

[1] G.N. Frederickson, *The information theory bound is tight for selection in a heap*, Proceedings of the $22^{nd}$ Annual ACM Symposium on Theory of Computing, p. 26–33, 1990.
[2] D.E. Knuth, *The art of computer programming, Volume 3, Sorting and searching*, Addison-Wesley, 1973.
[3] M.A. Weiss and J.K. Navlakha, *The distribution of keys in a binary heap*, p. 510–516 in LNCS **382**, Springer, 1989.

Leiden, June 1991